

INTERRUPT AND TRAP HANDLING IN AN EMBEDDED MULTI-THREAD  
PROCESSOR TO AVOID PRIORITY INVERSION AND MAINTAIN REAL-TIME  
OPERATION

Robert E. Ober

Roger D. Arnold

Daniel F. Martin

Erik K. Norden

ABSTRACT

A real-time, multi-threaded embedded system includes rules for handling traps and interrupts to avoid problems such as priority inversion and re-entrancy. By defining a global interrupt priority value for all active threads and only accepting interrupts having a priority higher than the interrupt priority value, priority inversion can be avoided. Switching to the same thread before any interrupt servicing, and disabling interrupts and thread switching during interrupt servicing can simplify the interrupt handling logic. By storing trap background data for traps and servicing traps only in their originating threads, trap traceability can be preserved. By disabling interrupts and thread switching during trap servicing, unintended trap re-entrancy and servicing disruption can be prevented.